AFRL-IF-RS-TR-2002-86
Final Technical Report
April 2002

CRITICAL ANALYSIS OF THE USE OF REDUNDANCY TO ACHIEVE SURVIVABILITY IN THE PRESENCE OF MALICIOUS ATTACKS

University of Wisconsin - Milwaukee

Sponsored by
Defense Advanced Research Projects Agency
DARPA Order No. F165/00

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

AIR FORCE RESEARCH LABORATORY
INFORMATION DIRECTORATE
ROME RESEARCH SITE
ROME, NEW YORK

This report has been reviewed by the Air Force Research Laboratory, Information Directorate, Public Affairs Office (IFOIPA) and is releasable to the National Technical Information Service (NTIS). At NTIS it will be releasable to the general public, including foreign nations.

AFRL-IF-RS-TR-2002-86 has been reviewed and is approved for publication.

APPROVED: Kevin A. Kwiat

KEVIN A. KWIAT
Project Engineer

FOR THE DIRECTOR:

WARREN H. DEBANY, Technical Advisor
Information Grid Division
Information Directorate

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 074-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	APRIL 2002	Final Apr 97 – May 99

4. TITLE AND SUBTITLE
CRITICAL ANALYSIS OF THE USE OF REDUNDANCY TO ACHIEVE SURVIVABILITY IN THE PRESENCE OF MALICIOUS ATTACKS

5. FUNDING NUMBERS
C - F30602-97-1-0205
PE - 62301E
PR - F165
TA - 40
WU - 23

6. AUTHOR(S)
Yvo Desmedt

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
University of Wisconsin – Milwaukee
Graduate School
Milwaukee Wisconsin 53201

8. PERFORMING ORGANIZATION REPORT NUMBER
N/A

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Defense Advanced Research Projects Agency	AFRL/IFGA
3701 North Fairfax Drive	525 Brooks Road
Arlington Virginia 22203-1714	Rome New York 13441-4505

10. SPONSORING / MONITORING AGENCY REPORT NUMBER
AFRL-IF-RS-TR-2002-86

11. SUPPLEMENTARY NOTES
AFRL Project Engineer: Kevin A. Kwiat/IFGA/(315) 330-1692/Kevin.Kwiat@rl.af.mil

12a. DISTRIBUTION / AVAILABILITY STATEMENT
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 Words)*
From the research on communication security one learns that, although redundancy has been utilized to achieve reliability, if the errors are caused maliciously, then the use of redundancy does not necessarily work. The goal is to adapt the lesson from the research on communication security to study when redundancy can and cannot be used to achieve survivability. Although this study was curtailed by the removal of funding, partial results were obtained for generalizing the attack to redundant computations of multiple inputs and creating an algorithm to identify the most critical tasks.

14. SUBJECT TERMS
Distributed Systems Survivability, Fault Tolerance, Security

15. NUMBER OF PAGES
9

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

TABLE OF CONTENTS

Objective

From the research on communication security one learns that, although redundancy has been utilized to achieve reliability, if the errors are caused maliciously the use of redundancy does not necessarily work. The goal is to adapt the lessons from the research on communication security to study when redundancy can and cannot be used to achieve survivability.

Approach

Redundancy has been used to achieve reliability in the context of fault tolerant computation, reliable communication and reliable networks. While reliability is solely concerned with accidental errors, survivability must also deal with malicious faults.

One can distinguish two types of malicious errors. In the first one, the faults are independent, while in the second they are dependent. Examples are now given to illustrate when these assumptions may be realistic in the context of protecting the survivability of computer systems. Suppose that the redundant hardware, algorithms and software used have been developed independently. Then the opponents likely need to develop independent attacks for many of these subsystems to be successful (except if a platform independent attack can be mounted). If, on the other hand, the same software has been replicated, a fault will be duplicated, which implies that the faults are dependent.

Now, when the malicious errors are independent it is reasonable to assume that when dealing with an attack with limited resources, the number of faults are limited. However, such an assumption makes no sense when the faults are strongly dependent (for example when the same faulty software has been replicated).

In the context of communication security, redundancy helps when dealing with a limited number of independent faults (using error-correcting codes), but the use of error-detection (or error-correcting) codes does not help when dealing with unlimited dependent errors. However, the use of authentication mechanisms allows one to detect the existence of an unlimited number of malicious faults. One should note that the work of Seberry and Safavi-Naini (see reference 1) has demonstrated that some authentication methods are nothing else than wrapped error-detection codes. The open problem whether and when redundancy helps to achieve survivability can, based on this analogy, be split into two subquestions depending whether the faults are dependent or not.

To answer the first subquestion - whether and when redundancy helps to achieve reliability when the faults (including Byzantine ones) are independent - several mathematical models have been developed. A directed multigraph based model and a monotone graph based model have been analyzed (see Accomplishments for details).

When viewing the input to the computation as the "sender" and the final output as a "receiver", one can link the problem of survivable computing with network security. Multiple-(vertex)-connected graphs have been used to achieve network reliability, for example. However, the algorithms developed in this context assume that all vertices (servers) know the graph, which is unrealistic in the scenario of wrapped servers. So, algorithms, if possible, to deal with the case the servers do not know this graph are being developed. One has already proven that these algorithms cannot be extended to a directed multi-graph case (see Accomplishments for details).

A main part of the second subquestion is whether replicated computation, possibly faulty, can be wrapped in such a way that one can detect an unlimited number of dependent faults. It is known that this is possible in the communication security context, using authentication methods. (This problem was to be studied during the third year of the project, in the context of a very general study on the impact of redundancy to achieve survivability in a malicious environment.

Accomplishments

Accomplishment 1

Modeling a scenario is which the adversary is malicious should allow for a dynamic topology in which changes in the system may take place without the (non-faulty) processors being aware of it. It should also allow for the most general type of processor which could represent a simple gate, a software package, or a powerful computer. So memory and the ability to perform complicated operations must be allowed for. The model should also describe the structure of the system at the appropriate level of abstraction: it must distinguish those aspects which are relevant to the computation and abstract out those aspects which are not essential. Such a model should offer the maximum flexibility to the designer. Previous models based on the traditional setting of computation theory are not suitable of our purpose.

Based on our analysis of redundant computation systems with multiple inputs, several models have been introduced and analyzed. Specifically, two models for independent faults were introduced: A directed multi-graph with colored edges model and a monotone graph model, and two models for dependent faults: A monotone graph with colored vertices and a monotone graph with partial orders on the colors of the vertices. More details have been given in the published paper.

A directed multi-graph with colored edges model: A redundant computation system can be modeled by a directed multi-graph with colored edges. There is at least one input vertex and one output vertex. One assumes that there is at most one edge of any given color which joins distinct vertices. There are several possible applications for this model. For example, processors whose inputs have the same color need only use one input (when there are no faults). If the colors are different then the processor must use one input for

each of the input colors, to carry out its computation (or whatever it is supposed to do). For example, the processors of the aviation control system need data from several sources such as the airplane's speed, position, and the processor can decide the airplane's speed by data from any one of the speed sensors, etc. Of course, this is only one of many possible applications.

A monotone graph model: A monotone graph is defined to be a directed graph with two types of vertices, labeled and-vertices and or-vertices. The graph must have at least one input (source) vertex and one output (sink) vertex. Input vertices may be regarded as and-vertices.

A monotone graph with colored vertices: A computation redundant system with dependent faults can be modeled by a monotone graph with colored vertices. The main advantage of monotone graphs with color vertices is that it models the dependent faults in an appropriate level and it is a more powerful mathematical tool for the study of dependent faults. There are several possible applications for this model. For example, the processors with the same standards could be marked with the same color and all computers which run Windows 95 could be marked with another color. And when a vertex fails, then all vertices with the same color will have the same failure probability.

A monotone graph with partial orders on the colors of the vertices: The monotone graphs with colored vertices reflect the dependent faults in a natural way. This model however does not focus on the faults which are weakly dependent on one another and therefore it does not describe some of the finer aspects of dependent faults. In this model one identifies different types of vertices by a color. A color could correspond with an operating system, or with the microprocessor used, etc. This operating system could be replicated and different replications correspond to different vertices in the model. In many instances there is a hierarchy on the type of failures. For example, if the hardware of a computer has a design flaw, all operating systems that require that hardware may also fail. Also, if the operating system fails all application programs requiring that operating system will fail. So one has an hierarchy of types of vertices. This additional aspect can (more generally) be expressed by using a partial ordering on the colors. So, such a redundant computation system can be modeled by a monotone graph with colored vertices which in addition has a partial order on the colors.

Accomplishment 2

The monotone graph model has been used to compare the design of reliable systems in computations with one type of input versus the case with multiple inputs. While there is a polynomial time algorithm for finding vertex disjoint paths in networks, our work shows that the equivalent problem in computation with multiple inputs is **NP**-hard. Whence dependable computation with multiple inputs is **NP**-hard. It follows that the general case redundancy may not help to achieve survivability assuming that **P** is not equal to **NP**.

Accomplishment 3

Byzantine type of attacks in the case the graph is unknown have been described in the proposal. The goal of this research is to study when these can be prevented. One assumes that the redundant computation can be modeled by a network.

In the case the sender knows the network, but the receiver does not, the attacks can easily be prevented. The sender basically sends (together with the message and other data) via all paths used the following pair of information: (description of the graph, the paths used). This result was recently published in Electronics Letters (see ref 2).

In the case each node has a public key and an edge in the unknown graph corresponds with a certificate of the public key the Byzantine type of attacks described in the proposal can also be prevented. An efficient algorithm has been described when the graph is **5/2k+1** connected, where k is the number of faulty nodes (when the graph is only **2k+1** connected an exponential time algorithm has also been found). Several measures are needed to prevent the attack to succeed. One of those is to prevent a malicious node to claim that non-existing nodes exit. This gives the impression that the graph is much larger than in reality. Round Robin was used to slow down the faulty processors to achieve this subgoal. The details of the algorithm are described in a submitted paper. This part of the research has also an impact on network security.

Accomplishment 4

In traditional reliability and survivability, used in reliable network design for example, one has the following result. If the adversary can destroy **k** vertices, one needs at least **k+1** vertices to obtain the desired output. Our result shows that in multi-input reliability, it is possible to protect against an adversary who can destroy **ck** vertices (**c** a constant) while having only a redundancy factor of **k** (see List of submitted publications).

There are other potential applications of the models discussed under Accomplishment 1. For example, these models may be used to identify the most critical tasks in redundant computations and to allocate the available resources to the most critical tasks. These models may also be used to analyze the flows in computation systems with multiple inputs and may eventually be used to analyze the performance of a manufacturing system.

Conclusion

At the time when their research and its impact on fault-tolerant computations was being planned, the funding and the period of performance for the grant were reduced. Curtailing the future funding and schedule of both, resulted in this research ending prematurely.

References:

1. “Error-Correcting Codes for Authentication and Subliminal Channels”, IEEE Transaction on Information Theory, IT-37(1), pp. 13-17, January 1991.

2. “Secure Communication in an Unknown Network with Byzantine Faults” Electronics Letters, Vol. 34, No. 8, pp. 741-742, 1998.
Authors: Mike Burmester and Yvo Desmedt

3. “NP - Hardness of Dependable Computation with Multiple Inputs” Submitted to the Fifth ACM Conference on Computer and Communications Security for publication April 3, 1998. Authors: Yongge Wang, Yvo Desmedt, and Mike Burmester

4. “Secure Communication in an Unknown Network using Certificates” Submitted to the Fifth ACM Conference on Computer and Communications Security for publication April 3, 1998. Authors: Mike Burmester and Yvo Desmedt